

FIG 1

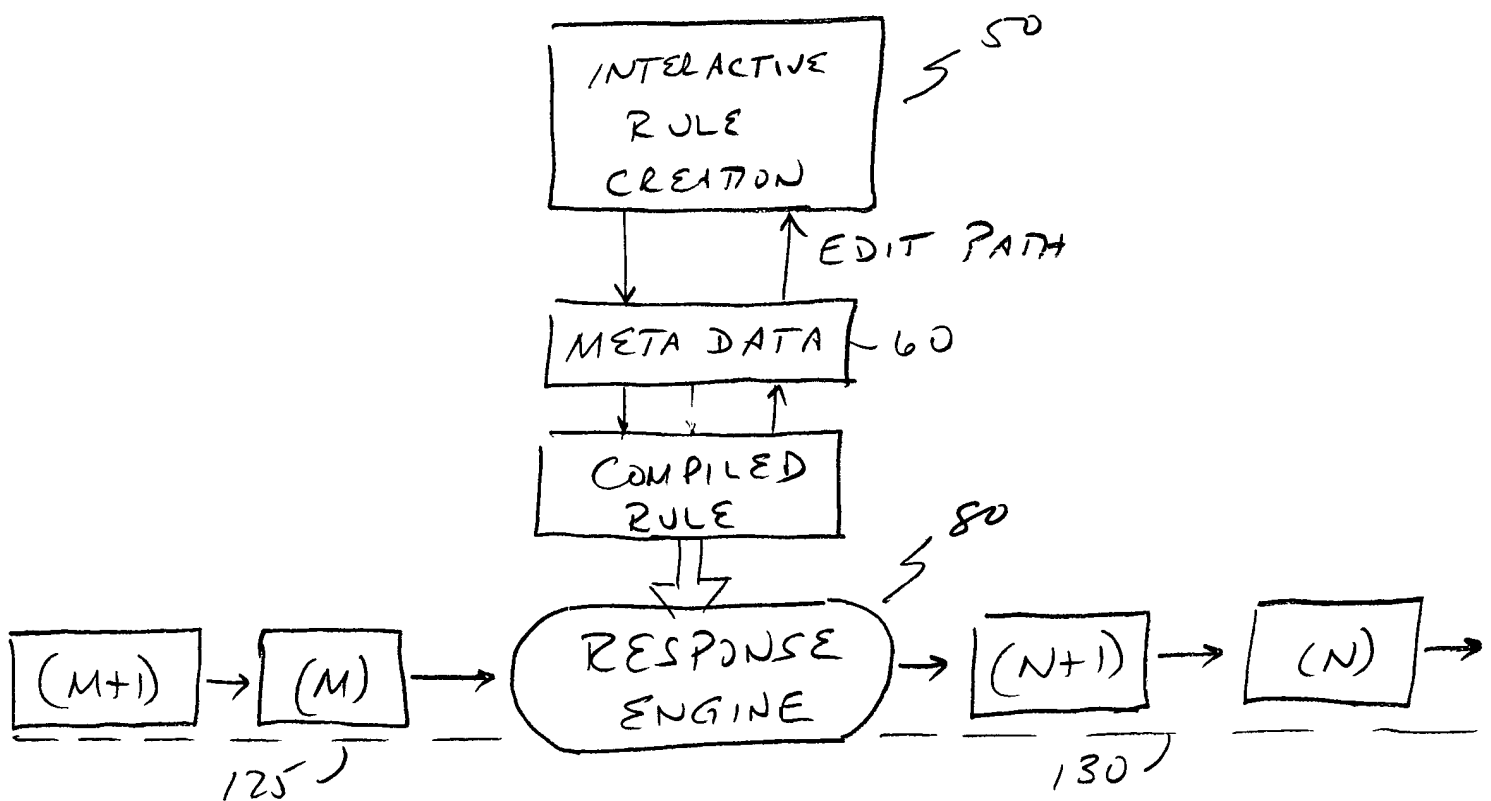


FIG 2

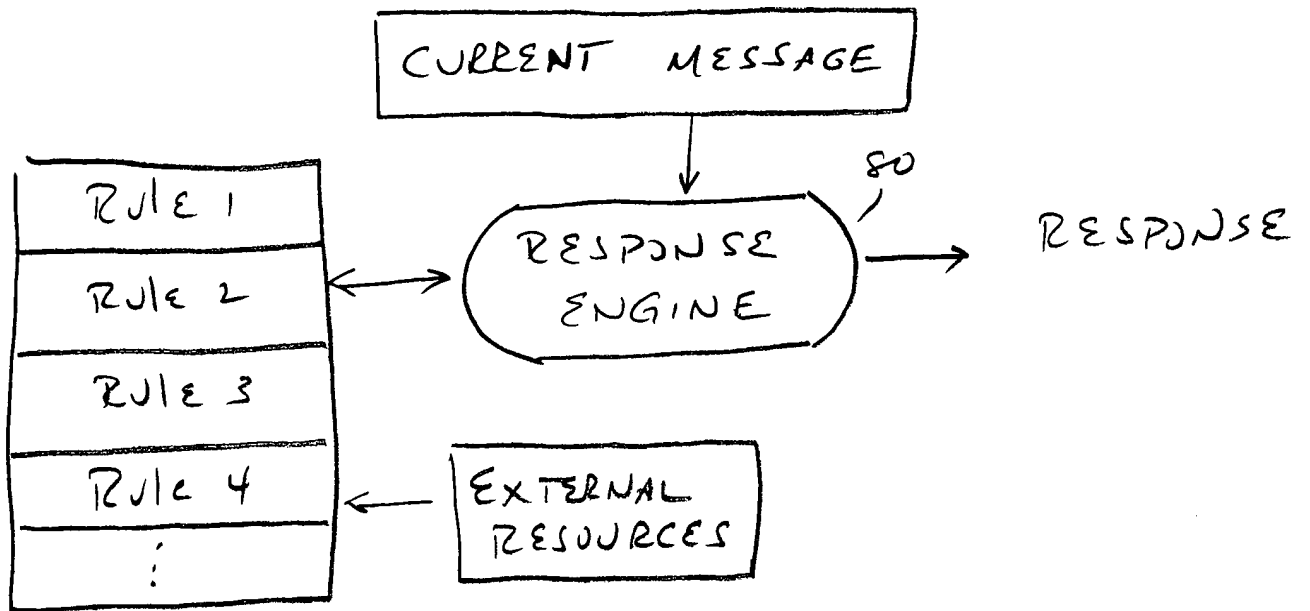


FIG 3

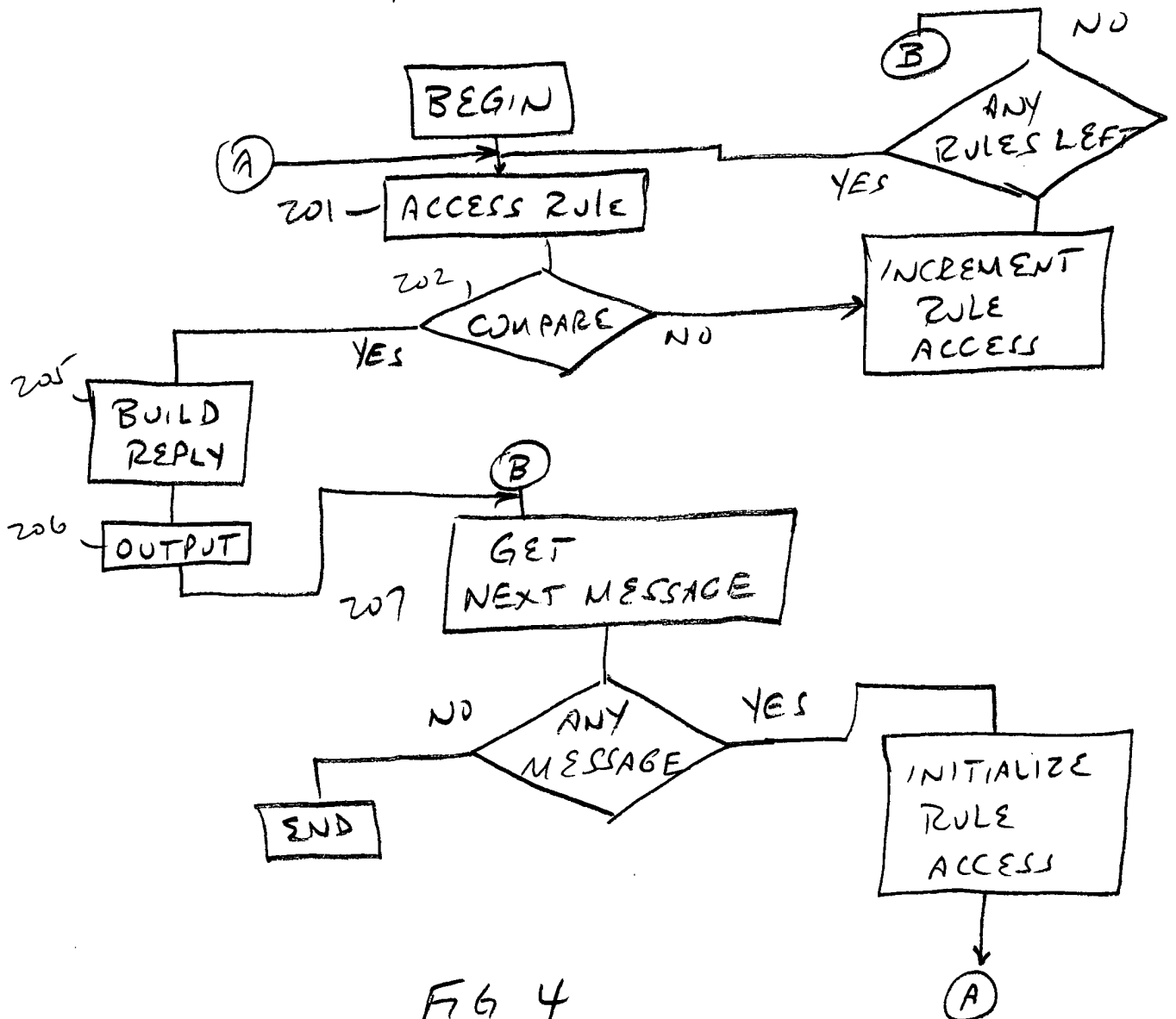


FIG 4

Create Comparison Wizard

You have chosen a comparison of type Complex Comparison

Raw Data Fixed Format Data XML Data

<IQShoppingRequest><Type>CREDITCHECK</Type><UserName>Dave</UserName><TotalPrice>44.94</TotalPrice></IQShoppingRequest>

Browse...

Start Offset

0

End Offset

0

AND if = <some value> then Add

Remove

Group

Ungroup

<- Back Next -> Options... Cancel ?

Fig 5

**Create Comparison Wizard**

You have chosen a comparison of type Complex Comparison

Raw Data Fixed Format Data XML Data

anishTemplate Template Management

Field Name	Value
ID1 (S:0 E:11)	<IQShopping
name (S:11 E:61)	Request><Type>CREDITCHECK</Type><UserName>Dave...
id (S:61 E:111)	erName><TotalPrice>44.94</TotalPrice></IQShoppingR
ADDITIONAL_DATA	equest>

AND if ID1 = <IQShopping then Add

Remove Group Ungroup

<- Back Next-> Options Cancel ?

F66

Create Comparison Wizard

You have chosen a comparison of type Complex Comparison

Raw Data Fixed Format Data XML Data

☐ <IQShoppingRequest>

- <Type> (CREDITCHECK)
- <UserName> (Dave)
- <TotalPrice> (44.94)

AND If ID1 = <IQShopping> then Add

Remove

Group

Ungroup

<- Back Next-> Options... Cancel ?

FIG 7

**Create Comparison Wizard** [X]

You have chosen a comparison of type **Complex Comparison**.

**Raw Data** | **Fixed Format Data** | **XML Data**

[-] **<IQShoppingRequest>**

- **<Type> (CREDITCHECK)**
- **<UserName> (Dave)**
- **<TotalPrice> (44.94)**

---

[AND] if **<TotalPrice>** **=** **44.94** then **Add**

If **<TotalPrice>** **AND** **OR**

**Selection**  
*Conjunction*

**Comparison**  
*Operation*

**Comparison**  
**Value**

**Remove**  
**Group**  
**Ungroup**

[- Back | Next -> | Options... | Cancel | ?

FG 8

Create Comparison Wizard

You have chosen a comparison of type Complex Comparison

Raw Data | Fixed Format Data | XML Data

☐ <IQShoppingRequest>

- <Type> (CREDITCHECK)
- <UserName> (Dave)
- <TotalPrice> (44.94)

AND If <TotalPrice> = 44.94 then Add

If <TotalPrice> = 44.94

Not =  
>  
>=  
<  
<=  
Contains  
Is Empty

Remove  
Group  
Ungroup

<- Back Next -> Options Cancel ?

FIG 9

Create Comparison Wizard

You have chosen a comparison of type Complex Comparison

Raw Data | Fixed Format Data | XML Data

☐ <IQShoppingRequest>

- ☐ <Type> (CREDITCHECK)
- ☐ <UserName> (Dave)
- ☐ <TotalPrice> (44.94)

OR if <UserName> = Dave then Add

if <TotalPrice> = 44.94  
AND (<Type> = CREDITCHECK  
OR <UserName> = Dave)

Remove  
Group  
Ungroup

<- Back Next-> Options... Cancel ?

FG 10



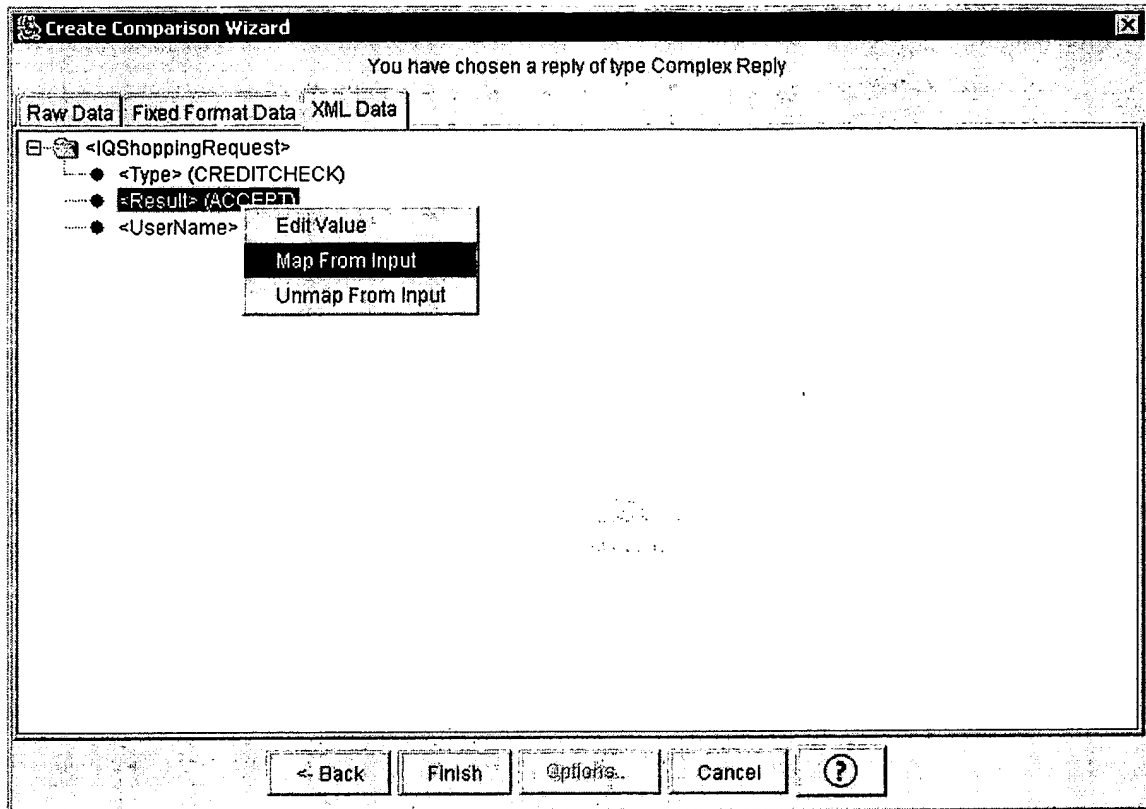


FIG 11

**Data Substitution**

Data Substitution Types: **Cache**

Parameter Name: <TotalPrice>

Current value: 44.94

**Configuration**

☒ Get from cache ☐ Send to cache

Name in cache:

**OK** **Cancel**

FIG 12

**Data Substitution**

Data Substitution Types: Database

Parameter Name: <TotalPrice>

Current value: 44.94

Configuration

SQL Statement DB Connection

Test Statement

OK Cancel

FG 13

**Data Substitution**

Data Substitution Types: **File**

Parameter Name: <TotalPrice>

Current value: 44.94

**Configuration**

File Name:  **Browse...**

Column:  ☒ Restart at end

Delimiter: ☒ Comma ☐ Tab

**OK** **Cancel**

Fig 14

Create Comparison Wizard - Raw

You have chosen a comparison of type Complex Comparison

Raw Data | Fixed Format Data | XML Data |

`<IQShoppingRequest><Type>CREDITCHECK</Type><UserName>Dave</UserName>  
<TotalPrice>44.94</TotalPrice></IQShoppingRequest>`

Load...

Start Offset

19

End Offset

25

AND if Start:19 End:25 = <Type> then Add

If Start:19 End:25 = <Type>

Remove

Group

Ungroup

<- Back Next -> Options Cancel ?

FIG 15a

```

- <Comparison baseClass="com.classiq.application.simulator.BaseComparison" className="Raw">
  <!-- This is the name of the comparison (may be different than the class name) -->
  <name>Raw</name>
  <!-- The delay used by the comparison -->
  <delay>0</delay>
  <!-- The remove after first use attribute -->
  <removeUsed>false</removeUsed>
- <!--
  This section describes the compare method. It contains 1 attribute:
  1) name - This is the name of the comparison type used. This is the label one w
-->
- <comparisonType name="Complex Comparison">
  <!-- These are fields used by the comparison, if any -->
  - <fields>
    - <templates>
      <template name="customer" />
    </templates>
  </fields>
  <!-- This is any information used for the compareInit method. Anything under this element
  can be unique for each type of comparison -->
  <init />
  <!-- This is any information used for the compare method. Anything under this element can
  be unique for each type of comparison -->
  - <method>
    - <ConditionData endParenIndex="0" leadingParens="" startParenIndex="0" theAddType="AND"
      theCompareType="" trailingParens="">
      - <theCompareValue>
        <![CDATA[ Type>CREDITCHECK</Ty ]]>
      </theCompareValue>
      - <theSelectionData>
        - <item>
          - <key>
            <![CDATA[ SELECTED_VALUE ]]>
          </key>
          - <value>
            <![CDATA[ Type>CREDITCHECK</Ty ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ END_OFFSET ]]>
          </key>
          - <value>
            <![CDATA[ 40 ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ TAG_NAME ]]>
          </key>
          - <value>
            <![CDATA[ last ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ VIEWER_TYPE ]]>
          </key>
          - <value>
            <![CDATA[ 2 ]]>
          </value>
        </item>
        - <item>
          - <key>

```

FG 1561

```

        <![CDATA[ TEMPLATE_NAME  ]]>
    </key>
- <value>
    <![CDATA[ customer  ]]>
</value>
</item>
- <item>
    - <key>
        <![CDATA[ START_OFFSET  ]]>
    </key>
    - <value>
        <![CDATA[ 20  ]]>
    </value>
</item>
</theSelectionData>
</ConditionData>
</method>
</comparisonType>
- <!--
    This section describes the reply method. It contains 1 attribute:
    1) name - This is the name of the reply type used. This is the label one would :

-->
- <replyType name="Complex Reply">
    <!-- These are fields used by the reply, if any -->
    - <fields>
        <templates />
    </fields>
    <!-- This is any information used for the replyInit method. Anything under this element can
    be unique for each type of reply -->
    - <init>
        <dir>file:/C:/Solstice/Integra/Enterprise/ApplicationSimulations/Comparisons/</dir>
        <file>Raw_out.txt</file>
    </init>
    <!-- This is any information used for the reply method. Anything under this element can be
    unique for each type of reply -->
    - <method>
        - <BaseReplyMessage>
            <![CDATA[ Select or enter a message  ]]>
        </BaseReplyMessage>
        <Mappings />
        <DataSubs />
    </method>
</replyType>
</Comparison>

```

FIG 1562

```

public class Raw extends
com.classiq.application.simulator.BaseComparison {
private long delay;
private boolean removeUsed;
//comparison variables
public static String[] compareTemplateNames = new String[]{"customer"} ;
//reply variables
private String strFileContent;
public static String[] replyTemplateNames = new String[]{} ;
private void comparisonInit() throws Exception {

}
private void replyInit() throws Exception {
java.net.URL url = new java.net.URL
(com.classiq.util.FileUtil.getComparisonResourceFileName(getName(),
"out"));
java.io.FileInputStream fis = new java.io.FileInputStream(url.getFile
());
int i;
strFileContent = "";
while((i = fis.read()) != -1) {
strFileContent += (char)i;
}
fis.close();

}
public long getDelay() { return delay; }
public void setDelay(long d) { delay = d; }
public boolean isDelaySettable() { return true; }
public boolean getRemoveUsed() { return removeUsed; }
public void setRemoveUsed(boolean b) { removeUsed = b; }
public boolean compareMessage(String incomingMessage) throws Exception {
if (ComplexComparisonUtil.compare
(com.classiq.message.parsers.parserUtil.getValueUsingTemplate
("customer","last",incomingMessage),"=", "Type>CREDITCHECK</Ty")
){
return true;
}
return false;
}
public String getReply(String incomingMessage) throws Exception {
String theReply = strFileContent;

return theReply;

}
public Raw() throws Exception {
delay = 0;
removeUsed = false;
setName("Raw");
comparisonInit();
replyInit();
}
}

```

FIG 15C



Create Comparison Wizard - Raw

You have chosen a comparison of type Complex Comparison

Raw Data

Fixed Format Data

XML Data

customer

Template Management ...

Field Name	Value
name (S:0 E:20)	<IQShoppingRequest><
last (S:20 E:40)	Type=CREDITCHECK=</Ty

AND

if

last

=

Type>CREDITCHECK</Ty

then

Add

if last = Type>CREDITCHECK</Ty

Remove

Group

Ungroup

<- Back

Next ->

Options

Cancel

F4 16c

```

- <Comparison baseClass="com.classiq.application.simulator.BaseComparison" className="Fixed">
  <!-- This is the name of the comparison (may be different than the class name) -->
  <name>Fixed</name>
  <!-- The delay used by the comparison -->
  <delay>0</delay>
  <!-- The remove after first use attribute -->
  <removeUsed>false</removeUsed>
- <!--
  This section describes the compare method. It contains 1 attribute:
  1) name - This is the name of the comparison type used. This is the label one w
-->
- <comparisonType name="Complex Comparison">
  <!-- These are fields used by the comparison, if any -->
  <fields>
    - <templates>
      <template name="customer" />
    </templates>
  </fields>
  <!-- This is any information used for the compareInit method. Anything under this element
  can be unique for each type of comparison -->
  <init />
  <!-- This is any information used for the compare method. Anything under this element can
  be unique for each type of comparison -->
- <method>
  - <ConditionData endParenIndex="0" leadingParens="" startParenIndex="0" theAddType="AND"
    theCompareType="=" trailingParens="">
    - <theCompareValue>
      <![CDATA[ <IQShoppingRequest>< ]]>
    </theCompareValue>
    - <theSelectionData>
      - <item>
        - <key>
          <![CDATA[ SELECTED_VALUE ]]>
        </key>
        - <value>
          <![CDATA[ <IQShoppingRequest>< ]]>
        </value>
      </item>
      - <item>
        - <key>
          <![CDATA[ END_OFFSET ]]>
        </key>
        - <value>
          <![CDATA[ 20 ]]>
        </value>
      </item>
      - <item>
        - <key>
          <![CDATA[ TAG_NAME ]]>
        </key>
        - <value>
          <![CDATA[ name ]]>
        </value>
      </item>
      - <item>
        - <key>
          <![CDATA[ VIEWER_TYPE ]]>
        </key>
        - <value>
          <![CDATA[ 2 ]]>
        </value>
      </item>
      - <item>
        - <key>

```

AG 1661

```

        <![CDATA[ TEMPLATE_NAME  ]]>
    </key>
    - <value>
        <![CDATA[ customer  ]]>
    </value>
</item>
- <item>
    - <key>
        <![CDATA[ START_OFFSET  ]]>
    </key>
    - <value>
        <![CDATA[ 0  ]]>
    </value>
</item>
</theSelectionData>
</ConditionData>
</method>
</comparisonType>
- <!--
    This section describes the reply method. It contains 1 attribute:
        1) name - This is the name of the reply type used. This is the label one would :

-->
- <replyType name="Complex Reply">
    <!-- These are fields used by the reply, if any -->
    - <fields>
        <templates />
    </fields>
    <!-- This is any information used for the replyInit method. Anything under this element can
    be unique for each type of reply -->
    - <init>
        <dir>file:/C:/Solstice/Integra/Enterprise/ApplicationSimulations/Comparisons/</dir>
        <file>Fixed_out.txt</file>
    </init>
    <!-- This is any information used for the reply method. Anything under this element can be
    unique for each type of reply -->
    - <method>
        - <BaseReplyMessage>
            <![CDATA[ Select or enter a message  ]]>
        </BaseReplyMessage>
        <Mappings />
        <DataSubs />
    </method>
</replyType>
</Comparison>

```

Fig 16b2

```

public class Fixed extends
com.classiq.application.simulator.BaseComparison {
private long delay;
private boolean removeUsed;
//comparison variables
public static String[] compareTemplateName = new String[]{"customer"} ;
//reply variables
private String strFileContent;
public static String[] replyTemplateName = new String[]{} ;
private void comparisonInit() throws Exception {

}
private void replyInit() throws Exception {
java.net.URL url = new java.net.URL
(com.classiq.util.FileUtil.getComparisonResourceFileName(getName(),
"out"));
java.io.FileInputStream fis = new java.io.FileInputStream(url.getFile
());
int i;
strFileContent = "";
while((i = fis.read()) != -1) {
strFileContent += (char)i;
}
fis.close();

}
public long getDelay() { return delay; }
public void setDelay(long d) { delay = d; }
public boolean isDelaySetable() { return true; }
public boolean getRemoveUsed() { return removeUsed; }
public void setRemoveUsed(boolean b) { removeUsed = b; }
public boolean compareMessage(String incomingMessage) throws Exception {
if (ComplexComparisonUtil.compare
(com.classiq.message.parsers.parserUtil.getValueUsingTemplate
("customer","name",incomingMessage),"=", "<IQShoppingRequest><")
){
return true;
}
return false;
}
public String getReply(String incomingMessage) throws Exception {
String theReply = strFileContent;

return theReply;

}
public Fixed() throws Exception {
delay = 0;
removeUsed = false;
setName("Fixed");
comparisonInit();
replyInit();
}
}

```

Fig 16C

Create Comparison Wizard - Raw

You have chosen a comparison of type Complex Comparison

Raw Data | Fixed Format Data | XML Data

☐ <IQShoppingRequest>

- ☐ <Type> (CREDITCHECK)
- ☐ <UserName> (Dave)
- ☒ <TotalPrice> (44.94)

AND if <TotalPrice> = 44.94 then Add

If <TotalPrice> = 44.94

Remove

Group

Ungroup

<- Back Next -> Options Cancel

FIG 17a

```

- <Comparison baseClass="com.classiq.application.simulator.BaseComparison" className="XML">
  <!-- This is the name of the comparison (may be different than the class name) -->
  <name>XML</name>
  <!-- The delay used by the comparison -->
  <delay>0</delay>
  <!-- The remove after first use attribute -->
  <removeUsed>false</removeUsed>
- <!--
  This section describes the compare method. It contains 1 attribute:
  1) name - This is the name of the comparison type used. This is the label one w
-->
- <comparisonType name="Complex Comparison">
  <!-- These are fields used by the comparison, if any -->
  - <fields>
    <templates />
  </fields>
  <!-- This is any information used for the compareInit method. Anything under this element
  can be unique for each type of comparison -->
  <init />
  <!-- This is any information used for the compare method. Anything under this element can
  be unique for each type of comparison -->
  - <method>
    - <ConditionData endParenIndex="0" leadingParens="" startParenIndex="0" theAddType="AND"
      theCompareType="=" trailingParens="">
      - <theCompareValue>
        <![CDATA[ 44.94 ]]>
      </theCompareValue>
      - <theSelectionData>
        - <item>
          - <key>
            <![CDATA[ SELECTED_VALUE ]]>
          </key>
          - <value>
            <![CDATA[ 44.94 ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ XML_PATH ]]>
          </key>
          - <value>
            <![CDATA[ [TotalPrice*0, IQShoppingRequest*0] ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ TAG_NAME ]]>
          </key>
          - <value>
            <![CDATA[ <TotalPrice> ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ VIEWER_TYPE ]]>
          </key>
          - <value>
            <![CDATA[ 3 ]]>
          </value>
        </item>
        - <item>
          - <key>
            <![CDATA[ XML_PATH_STRING ]]>
          </key>

```

FG 1761

```

- <value>
  <![CDATA[ [TotalPrice*0, IQShoppingRequest*0] ]]>
</value>
</item>
</theSelectionData>
</ConditionData>
</method>
</comparisonType>
- <!--
  This section describes the reply method. It contains 1 attribute:
  1) name - This is the name of the reply type used. This is the label one would
-->

-->
- <replyType name="Complex Reply">
  <!-- These are fields used by the reply, if any -->
  <fields>
    <templates />
  </fields>
  <!-- This is any information used for the replyInit method. Anything under this element can
  be unique for each type of reply -->
  <init>
    <dir>file:/C:/Solstice/Integra/Enterprise/ApplicationSimulations/Comparisons/</dir>
    <file>XML_out.txt</file>
  </init>
  <!-- This is any information used for the reply method. Anything under this element can be
  unique for each type of reply -->
  <method>
    <BaseReplyMessage>
      <![CDATA[ Select or enter a message ]]>
    </BaseReplyMessage>
    <Mappings />
    <DataSubs />
  </method>
</replyType>
</Comparison>

```

Fig 1762

```

public class XML extends
com.classiq.application.simulator.BaseComparison {
private long delay;
private boolean removeUsed;
//comparison variables
public static String[] compareTemplateName = new String[]{} ;
//reply variables
private String strFileContent;
public static String[] replyTemplateName = new String[]{} ;
private void comparisonInit() throws Exception {

}
private void replyInit() throws Exception {
java.net.URL url = new java.net.URL
(com.classiq.util.FileUtil.getComparisonResourceFileName(getName(),
"out"));
java.io.FileInputStream fis = new java.io.FileInputStream(url.getFile
());
int i;
strFileContent = "";
while((i = fis.read()) != -1) {
strFileContent += (char)i;
}
fis.close();

}
public long getDelay() { return delay; }
public void setDelay(long d) { delay = d; }
public boolean isDelaySettable() { return true; }
public boolean getRemoveUsed() { return removeUsed; }
public void setRemoveUsed(boolean b) { removeUsed = b; }
public boolean compareMessage(String incomingMessage) throws Exception {
if (ComplexComparisonUtil.compare
(com.classiq.message.parsers.parserUtil.getValue(incomingMessage,new
String[]{"TotalPrice*0","IQShoppingRequest*0",}), "=", "44.94")
){
return true;
}
return false;
}
public String getReply(String incomingMessage) throws Exception {
String theReply = strFileContent;

return theReply;

}
public XML() throws Exception {
delay = 0;
removeUsed = false;
setName("XML");
comparisonInit();
replyInit();
}
}

```

Fig 17c



11/24/00

FIG 18

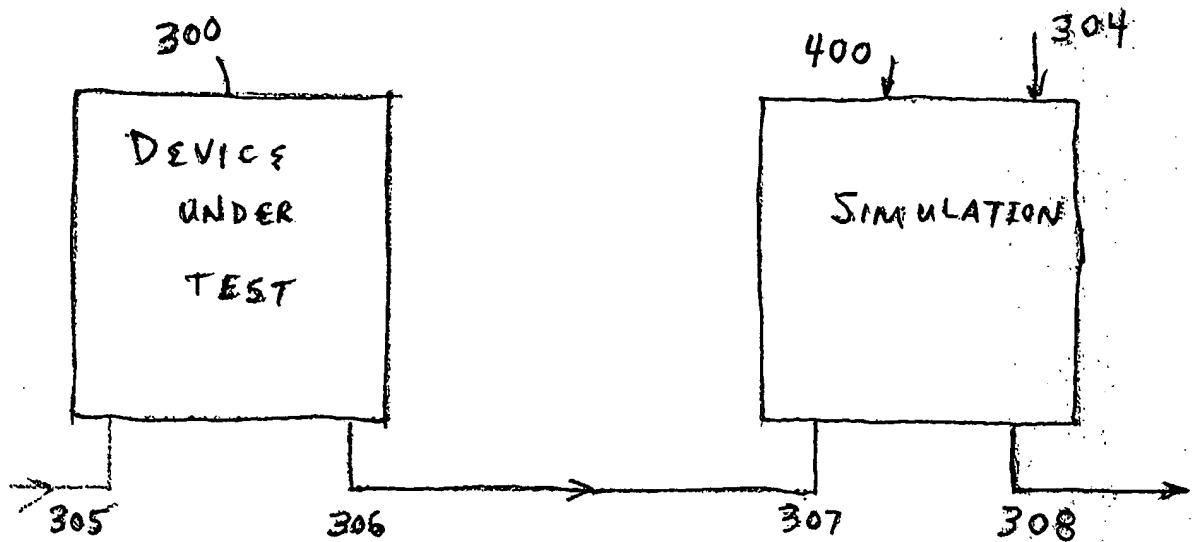


FIG 19

